

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN 3
LINEAR REGRESSION

GV hướng dẫn	: Nguyễn Văn Quang Huy
	: Nguyễn Ngọc Toàn
Sinh viên thực hiện	: Lê Thị Hồng Hạnh
MSSV	: 22127103
Lớp	: 22CLC07

MỤC LỤC

I.	Các thư viện được sử dụng.....	2
II.	Mô tả cách làm	2
1.	Hàm của class <code>OLSLinearRegression</code>	2
2.	Hàm tính sai số tuyệt đối trung bình MAE	3
3.	Hàm của thư viện <code>seaborn</code> và <code>matplotlib.pyplot</code>	4
4.	Phân tích khám phá dữ liệu	6
5.	Xây dựng mô hình dự đoán chỉ số thành tích sử dụng toàn bộ 5 đặc trưng	13
6.	Xây dựng mô hình sử dụng 1 đặc trưng, tìm mô hình cho kết quả tốt nhất	14
7.	Xây dựng/thiết kế mô hình, tìm mô hình cho kết quả tốt nhất.....	17
III.	Tài liệu tham khảo	19

I. Các thư viện được sử dụng

- pandas: đọc dữ liệu từ file train.csv và test.csv
- numpy: tính toán, thao tác trên ma trận
- seaborn: hỗ trợ vẽ biểu đồ thống kê
- matplotlib.pyplot: vẽ cùng lúc nhiều biểu đồ trong cùng một figure

II. Mô tả cách làm

1. Hàm của class **OLSLinearRegression**

Lớp này dùng cho việc xử lý phương trình hồi quy tuyến tính sử dụng phương pháp bình phương tối thiểu

a. *def fit(self, X, y)*

Input:

- X (numpy.ndarray): ma trận đầu vào của model
- y (numpy.ndarray): ma trận đầu ra của model

Output:

- self: một instance của lớp OLSLinearRegression

Description:

- Hàm dùng để giải phương trình $Xw = y$ với $w = (X^T X)^{-1} X^T y$

b. *def get_params(self)*

Output:

- self.w (numpy.ndarray): hệ số của model

Description:

- Hàm trả về hệ số của model w đã được tính ở hàm fit()

c. *def predict(self, X)*

Input:

- X (numpy.ndarray): ma trận dữ liệu đầu vào của model

Output:

- y (numpy.ndarray): ma trận dữ liệu đầu ra của model

Description:

- Từ ma trận dữ liệu đầu vào X và hệ số của model w đã tính được, ta dự đoán ma trận dữ liệu đầu ra y bằng công thức $Xw = y$

2. Hàm tính sai số tuyệt đối trung bình MAE

- `def mae(y, y_hat)`

Input:

- `y` (numpy.ndarray): dữ liệu đầu ra của bộ dữ liệu
- `y_hat` (numpy.ndarray): dữ liệu đầu ra được dự đoán

Output:

- `mae` (float): sai số tuyệt đối trung bình

Description:

- Sai số tuyệt đối trung bình được tính bằng công thức $MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|$

3. Hàm của thư viện seaborn và matplotlib.pyplot [1] [2]

a. Hàm pandas.read_csv()

Input:

- Filename (string): tên cần đọc dữ liệu

Output:

- Bộ dữ liệu dưới dạng DataFrame

Description:

- Tiến hành đọc file ‘train.csv’ và ‘test.csv’ để lấy dữ liệu train và test

b. Hàm pandas.DataFrame.corr()

Return:

- Ma trận chỉ số tương quan giữa các đặc trưng

Description:

- Hàm dùng để tính mối tương quan theo từng cặp của các đặc trưng, ma trận tương quan được sử dụng để vẽ biểu đồ tương quan

c. Hàm seaborn.heatmap()

Input:

- DataFrame.corr(): ma trận mối tương quan giữa các đặc trưng
- annot (True): để hiển thị giá trị tương quan lên biểu đồ
- cmap (‘coolwarm’): bảng màu nóng lạnh cho biểu đồ tương quan

Description:

- Từ ma trận tương quan, hàm sẽ vẽ biểu đồ tương quan giữa các đặc trưng, hiển thị màu sắc giúp dễ dàng hơn cho việc phân tích dữ liệu

d. Hàm plt.subplots()

Input:

- nrows, ncols: số dòng, cột của figure
- figsize: kích thước của figure

Output:

- fig: một đối tượng figure quản lý các Axes
- ax: một hay nhiều đối tượng Axes

Description:

- Hàm dùng để vẽ nhiều biểu đồ cùng lúc trên cùng một hình

e. Hàm `seaborn.countplot()`

Input:

- data (DataFrame): dữ liệu cần thống kê
- x, y, hue: tên cột dữ liệu trong tập dữ liệu
- ax (Axes): đối tượng Axes để xác định thứ tự vẽ biểu đồ

Description:

- Hàm dùng để vẽ biểu đồ dạng bar, giúp thống kê số lượng sinh viên với mỗi mức giá trị.

f. Hàm `seaborn.lineplot()`

Input:

- data (DataFrame): dữ liệu cần thống kê
- x, y: tên cột dữ liệu trong tập dữ liệu
- ax (Axes): đối tượng Axes để xác định thứ tự vẽ biểu đồ

Description:

- Hàm dùng để vẽ biểu đồ đường thể hiện sự biến thiên giá trị của các đặc trưng và thành tích

g. Hàm `pandas.DataFrame.groupby().mean()`

Input:

- Label: Tên cột cần nhóm các giá trị

Description:

- Hàm để nhóm lại và tính trung bình điểm Performance Index của từng mức giá trị .

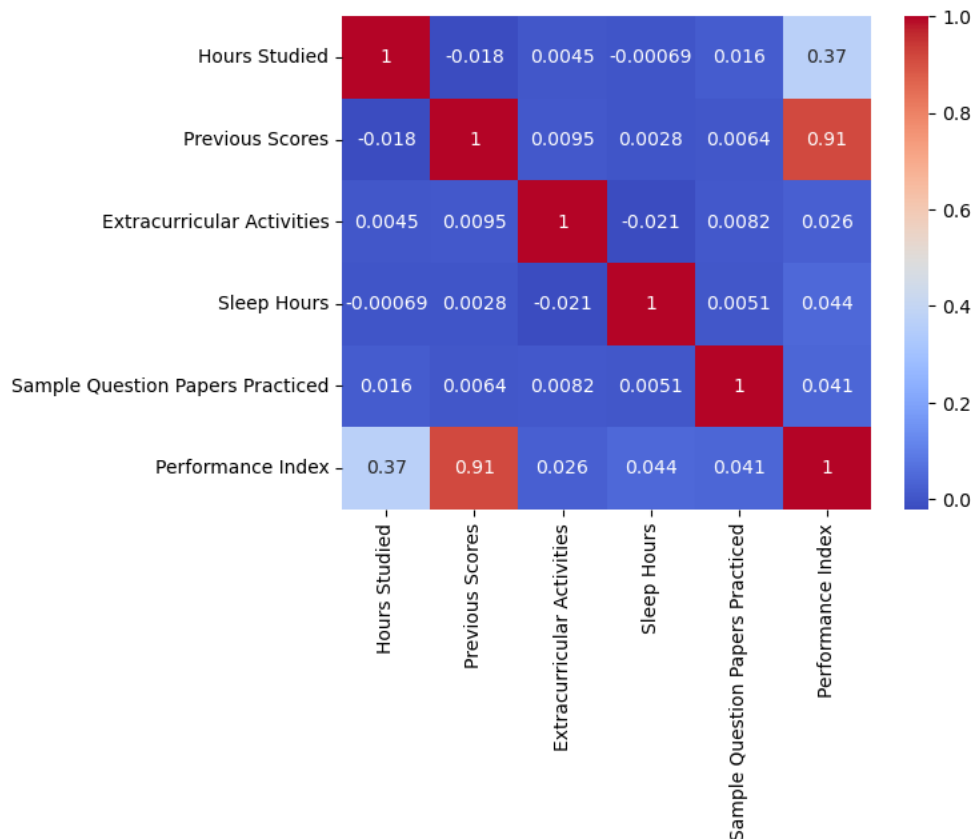
4. Phân tích khám phá dữ liệu

a. Phân tích hệ số tương quan

Hệ số tương quan [3] là chỉ số thống kê đo lường mức độ mạnh yếu của mối quan hệ giữa hai biến số. Trong đó:

- Hệ số tương quan có giá trị từ -1.0 đến 1.0. Kết quả được tính ra lớn hơn 1.0 hoặc nhỏ hơn -1 có nghĩa là có lỗi trong phép đo tương quan.
- Hệ số tương quan có giá trị âm cho thấy hai biến có mối quan hệ nghịch biến hoặc tương quan âm (nghịch biến tuyệt đối khi giá trị bằng -1)
- Hệ số tương quan có giá trị dương cho thấy mối quan hệ đồng biến hoặc tương quan dương (đồng biến tuyệt đối khi giá trị bằng 1)
- Tương quan bằng 0 cho hai biến độc lập với nhau.

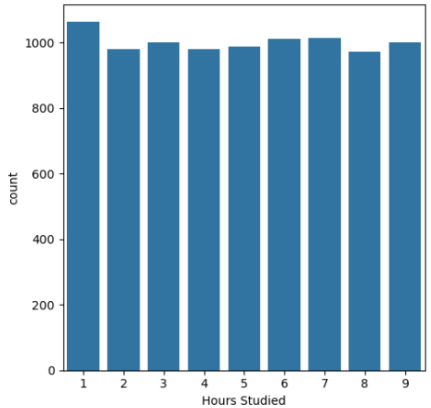
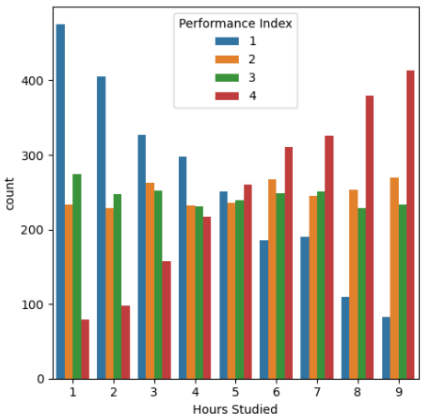
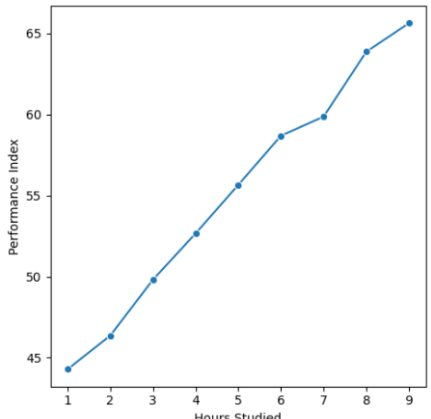
Sử dụng hàm heatmap() của thư viện seaborn để vẽ biểu đồ thể hiện sự tương quan giữa các đặc trưng với thành tích của sinh viên. Ta thấy được Hours Studied và Previous Scores là 2 đặc trưng có ảnh hưởng tích cực nhất đến Performance Index. Các đặc trưng còn lại đều có hệ số tương quan dương đối với Performance Index, tức chúng cũng có ảnh hưởng nhất định đến thành tích sinh viên nhưng giá trị nhỏ hơn nhiều so với hai đặc trưng kia.0



b. Phân tách dữ liệu Performance Index thành các khoảng

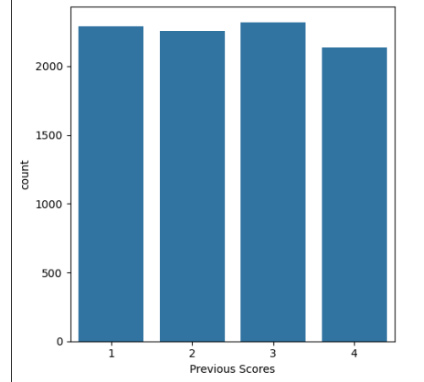
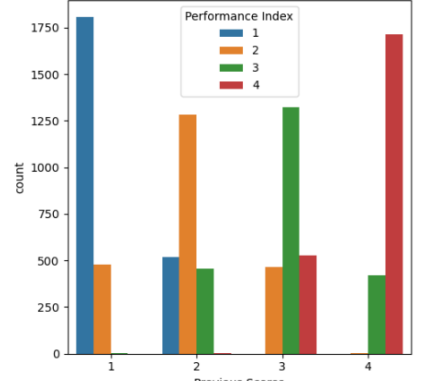
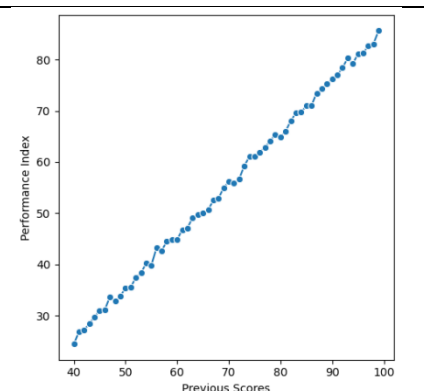
- Do giá trị của Performance Index nằm rải rác trong khoảng [10, 100], việc thống kê và phân tích đôi khi sẽ không thuận lợi do có quá nhiều giá trị trên trục tung hoặc hoành. Vì vậy, ta phân tách dữ liệu thành 4 mức điểm số là 1, 2, 3, 4 trong đó điểm ở các mức thỏa $1 < 2 < 3 < 4$. Sử dụng hàm `pandas.qcut()` với đối số truyền vào là cột dữ liệu Performance Index, số lượng mức muốn phân tách là 4 và label của từng khoảng là '1', '2', '3', '4'.
- Đối với mỗi đặc trưng, ta tiến hành thống kê số lượng của mỗi giá trị xuất hiện trong bộ dữ liệu, số lượng giá trị tương ứng với từng mức 1, 2, 3, 4 của Performance Index và biến thiên Performance Index trung bình của từng loại giá trị.
- Do đó, với mỗi đặc trưng ta sẽ tạo ra 3 subplots bằng hàm `plot.subplots` để thực hiện 3 thống kê phân tích nêu trên.

c. Phân tích đặc trưng Hours Studied

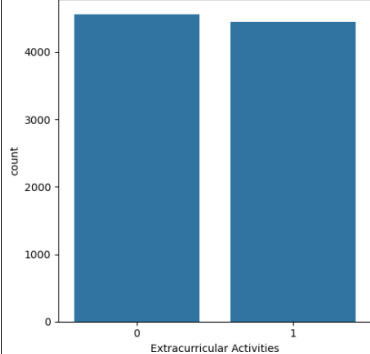
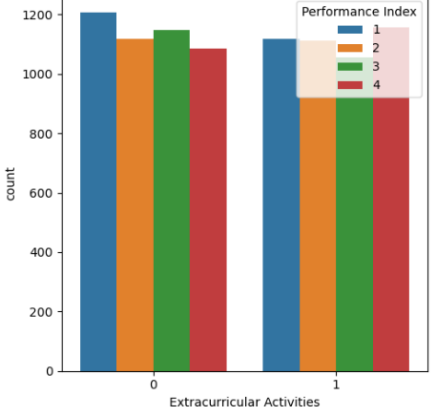
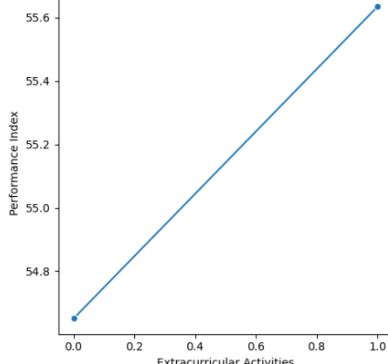
<p>Thống kê số lượng sinh viên theo giờ học</p> <p>Ta thấy được phân bố số lượng sinh viên của từng mức giờ học khá đồng đều trong bộ dữ liệu, không có giá trị giờ học nào có số lượng sinh viên ít hay nhiều quá vượt trội so với các giờ học còn lại</p>	 <table><tr><th>Hours Studied</th><th>count</th></tr><tr><td>1</td><td>1050</td></tr><tr><td>2</td><td>980</td></tr><tr><td>3</td><td>1000</td></tr><tr><td>4</td><td>980</td></tr><tr><td>5</td><td>990</td></tr><tr><td>6</td><td>1010</td></tr><tr><td>7</td><td>1010</td></tr><tr><td>8</td><td>980</td></tr><tr><td>9</td><td>1000</td></tr></table>	Hours Studied	count	1	1050	2	980	3	1000	4	980	5	990	6	1010	7	1010	8	980	9	1000																														
Hours Studied	count																																																		
1	1050																																																		
2	980																																																		
3	1000																																																		
4	980																																																		
5	990																																																		
6	1010																																																		
7	1010																																																		
8	980																																																		
9	1000																																																		
<p>Thống kê số lượng sinh viên theo giờ học và thành tích ở các mức 1, 2, 3, 4</p> <p>Ta thấy được rõ rệt số giờ học càng cao thì số lượng sinh viên đạt thành tích ở mức điểm số cao (mức 4) càng cao và số giờ học càng thấp thì số lượng sinh viên đạt điểm số thấp (mức 1) càng cao. Số lượng đạt điểm mức 2, 3 không quá dao động khi số giờ học biến thiên. Do đó, số giờ học có ảnh hưởng lớn đến số lượng sinh viên đạt mức điểm 1 và 4 nhiều nhất.</p>	 <table><tr><th>Hours Studied</th><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><td>1</td><td>450</td><td>250</td><td>280</td><td>80</td></tr><tr><td>2</td><td>410</td><td>230</td><td>250</td><td>100</td></tr><tr><td>3</td><td>330</td><td>260</td><td>250</td><td>160</td></tr><tr><td>4</td><td>300</td><td>230</td><td>220</td><td>220</td></tr><tr><td>5</td><td>250</td><td>240</td><td>240</td><td>260</td></tr><tr><td>6</td><td>190</td><td>270</td><td>250</td><td>310</td></tr><tr><td>7</td><td>190</td><td>250</td><td>240</td><td>330</td></tr><tr><td>8</td><td>110</td><td>250</td><td>230</td><td>380</td></tr><tr><td>9</td><td>80</td><td>270</td><td>240</td><td>410</td></tr></table>	Hours Studied	1	2	3	4	1	450	250	280	80	2	410	230	250	100	3	330	260	250	160	4	300	230	220	220	5	250	240	240	260	6	190	270	250	310	7	190	250	240	330	8	110	250	230	380	9	80	270	240	410
Hours Studied	1	2	3	4																																															
1	450	250	280	80																																															
2	410	230	250	100																																															
3	330	260	250	160																																															
4	300	230	220	220																																															
5	250	240	240	260																																															
6	190	270	250	310																																															
7	190	250	240	330																																															
8	110	250	230	380																																															
9	80	270	240	410																																															
<p>Điểm trung bình của sinh viên ở các mức giờ học</p> <p>Tại mỗi mức giờ học, ta tiến hành tính điểm trung bình của tất cả sinh viên có số giờ học đó. Có 9 giá trị giờ học nên thu được 9 giá trị điểm trung bình. Biểu diễn các giá trị đó bằng đồ thị line. Ta thấy số giờ học càng tăng thì điểm trung bình cũng tăng theo, do đó càng làm rõ thêm nhận định đặc trưng Hours Studied có ảnh hưởng đáng kể đến Performance Index.</p>	 <table><tr><th>Hours Studied</th><th>Performance Index</th></tr><tr><td>1</td><td>44</td></tr><tr><td>2</td><td>46</td></tr><tr><td>3</td><td>49</td></tr><tr><td>4</td><td>53</td></tr><tr><td>5</td><td>56</td></tr><tr><td>6</td><td>59</td></tr><tr><td>7</td><td>60</td></tr><tr><td>8</td><td>64</td></tr><tr><td>9</td><td>66</td></tr></table>	Hours Studied	Performance Index	1	44	2	46	3	49	4	53	5	56	6	59	7	60	8	64	9	66																														
Hours Studied	Performance Index																																																		
1	44																																																		
2	46																																																		
3	49																																																		
4	53																																																		
5	56																																																		
6	59																																																		
7	60																																																		
8	64																																																		
9	66																																																		

d. Phân tích đặc trưng Previous Scores

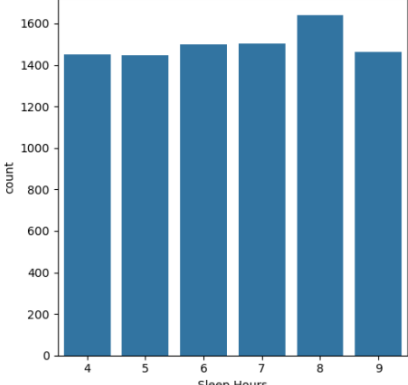
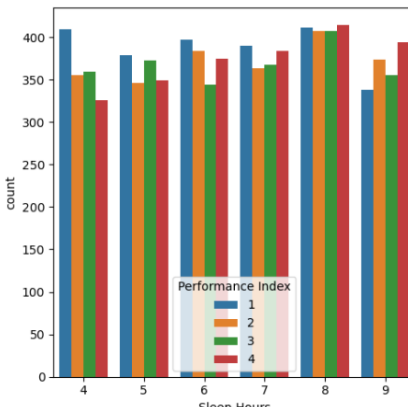
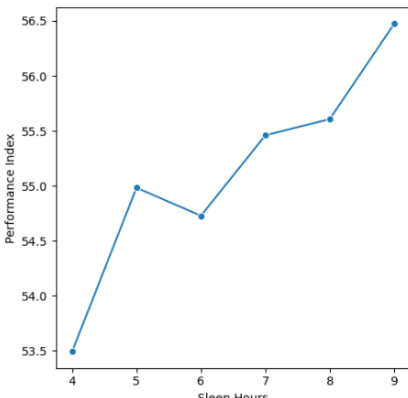
Do giá trị của Previous Scores cũng giống như Performance Index, việc thống kê và phân tích đôi khi sẽ không thuận lợi. Do đó, ta phân tách dữ liệu thành 4 mức điểm số là 1, 2, 3, 4 trong đó điểm ở các mức thỏa $1 < 2 < 3 < 4$. Sử dụng hàm `pandas.qcut()` với đối số truyền vào là cột dữ liệu Previous Scores, số lượng mức muốn phân tách là 4 và label của từng khoảng là '1', '2', '3', '4'.

<h3>Thống kê số lượng sinh viên theo từng mức Previous Scores</h3> <p>Ta thấy được phân bố số lượng sinh viên của từng mức điểm đạt được trước đó khá đồng đều trong bộ dữ liệu, không có mức điểm nào có số lượng sinh viên ít hay nhiều quá vượt trội so với các mức còn lại</p>	 <table><tr><th>Previous Scores</th><th>count</th></tr><tr><td>1</td><td>2100</td></tr><tr><td>2</td><td>2050</td></tr><tr><td>3</td><td>2150</td></tr><tr><td>4</td><td>2050</td></tr></table>	Previous Scores	count	1	2100	2	2050	3	2150	4	2050															
Previous Scores	count																									
1	2100																									
2	2050																									
3	2150																									
4	2050																									
<h3>Thống kê số lượng sinh viên theo số điểm đạt được ở các bài kiểm tra trước và thành tích cuối cùng ở các mức 1, 2, 3, 4</h3> <p>Đối với sinh viên chỉ đạt mức 1 ở Previous Scores thì gần như không có sinh viên nào đạt điểm mức 3, 4 ở Performance Index. Còn đối với các sinh viên có Previous Scores ở mức 4, hầu hết đều đạt điểm mức cao của Performance Index. Vì vậy, Previous Scores là một đặc trưng quan trọng để dự đoán Performance Index.</p>	 <table><tr><th>Previous Scores</th><th>Performance Index 1</th><th>Performance Index 2</th><th>Performance Index 3</th><th>Performance Index 4</th></tr><tr><td>1</td><td>1750</td><td>500</td><td>0</td><td>0</td></tr><tr><td>2</td><td>500</td><td>1300</td><td>450</td><td>0</td></tr><tr><td>3</td><td>0</td><td>450</td><td>1350</td><td>550</td></tr><tr><td>4</td><td>0</td><td>0</td><td>400</td><td>1700</td></tr></table>	Previous Scores	Performance Index 1	Performance Index 2	Performance Index 3	Performance Index 4	1	1750	500	0	0	2	500	1300	450	0	3	0	450	1350	550	4	0	0	400	1700
Previous Scores	Performance Index 1	Performance Index 2	Performance Index 3	Performance Index 4																						
1	1750	500	0	0																						
2	500	1300	450	0																						
3	0	450	1350	550																						
4	0	0	400	1700																						
<h3>Điểm trung bình của sinh viên ở các mức điểm Previous Scores</h3> <p>Tại giá trị Previous Scores, ta tiến hành tính điểm trung bình của tất cả sinh viên có Previous Scores đó. Nhìn chung, khi Previous Scores tăng thì Performance Index cũng tăng theo và sự tăng này ổn định. Kết luận, Previous Scores là một đặc trưng ảnh hưởng rất rõ ràng đến Performance Index.</p>	 <table><tr><th>Previous Scores</th><th>Performance Index</th></tr><tr><td>40</td><td>25</td></tr><tr><td>50</td><td>35</td></tr><tr><td>60</td><td>45</td></tr><tr><td>70</td><td>55</td></tr><tr><td>80</td><td>65</td></tr><tr><td>90</td><td>75</td></tr><tr><td>100</td><td>85</td></tr></table>	Previous Scores	Performance Index	40	25	50	35	60	45	70	55	80	65	90	75	100	85									
Previous Scores	Performance Index																									
40	25																									
50	35																									
60	45																									
70	55																									
80	65																									
90	75																									
100	85																									

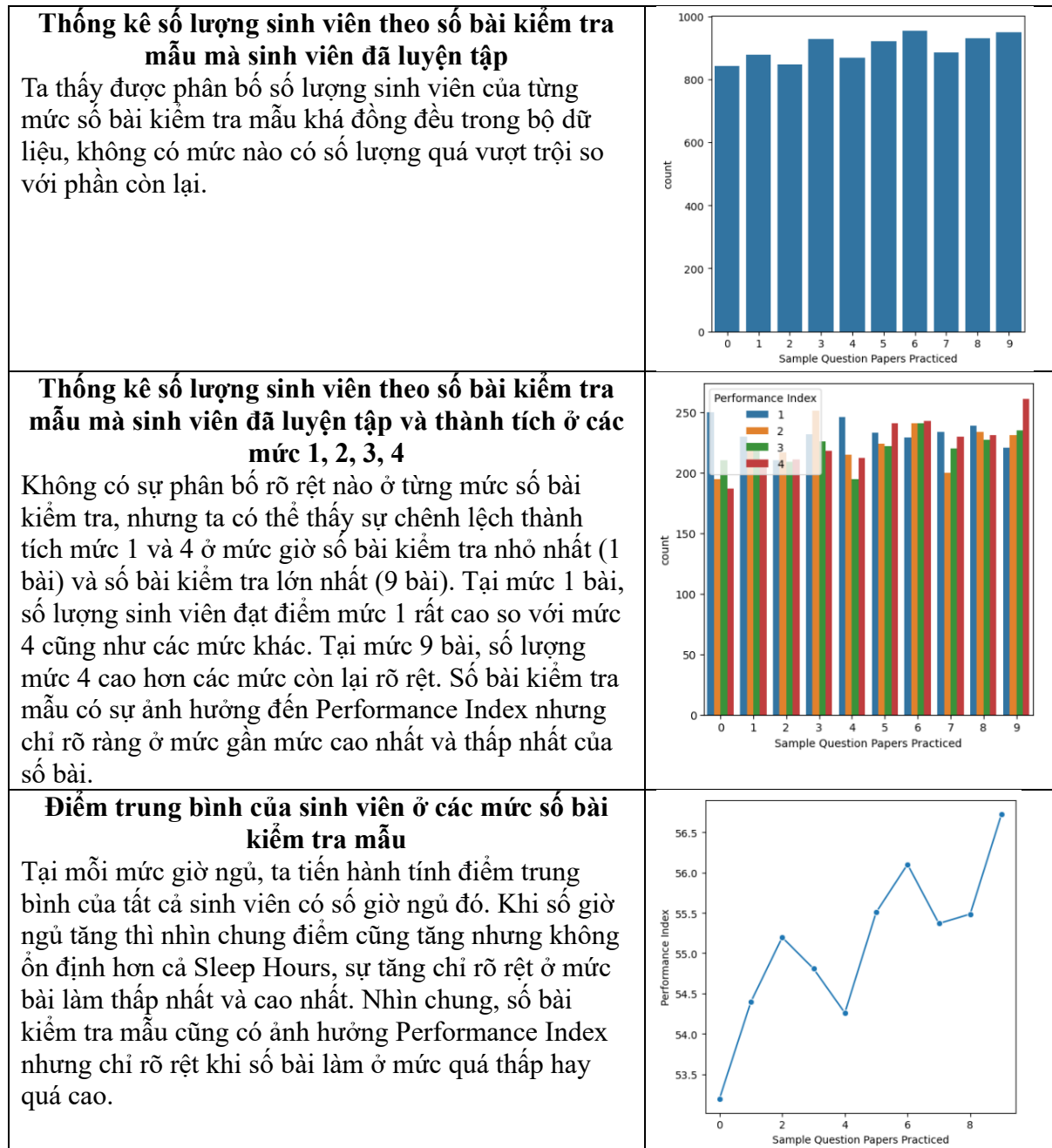
e. Phân tích đặc trưng Extracurricular Activities

<p>Thống kê số lượng sinh viên có hay không tham gia Extracurricular Activities</p> <p>Ta thấy được phân bố số lượng sinh viên có tham gia và không tham gia hoạt động ngoại khóa đồng đều trong bộ dữ liệu. Số lượng sinh viên có và không tham gia không có chênh lệch quá lớn.</p>	 <table><caption>Data for Extracurricular Activities Count</caption><thead><tr><th>Extracurricular Activities</th><th>count</th></tr></thead><tbody><tr><td>0</td><td>~4500</td></tr><tr><td>1</td><td>~4500</td></tr></tbody></table>	Extracurricular Activities	count	0	~4500	1	~4500									
Extracurricular Activities	count															
0	~4500															
1	~4500															
<p>Thống kê số lượng sinh viên theo hai trường hợp có hoặc không tham gia hoạt động và thành tích ở các mức 1, 2, 3, 4</p> <p>Phân bố các mức điểm của hai nhóm sinh viên có và không tham gia hoạt động không có quá nhiều chênh lệch, nhất là ở mức 2, 3. Đối với nhóm không tham gia thì số lượng thành tích mức 1 cao và mức 4 thấp một chút. Ngược lại, nhóm có tham gia có số lượng thành tích ở mức 1 thấp hơn và mức 4 cao hơn chút. Việc có hay không tham gia hoạt động có ảnh hưởng không quá lớn đến Performance Index, nhưng sẽ có sự chênh lệch nhỏ giữa mức điểm 1 và 4.</p>	 <table><caption>Data for Performance Index by Extracurricular Activities</caption><thead><tr><th>Extracurricular Activities</th><th>Performance Index 1</th><th>Performance Index 2</th><th>Performance Index 3</th><th>Performance Index 4</th></tr></thead><tbody><tr><td>0</td><td>~1200</td><td>~1100</td><td>~1150</td><td>~1050</td></tr><tr><td>1</td><td>~1100</td><td>~900</td><td>~950</td><td>~1000</td></tr></tbody></table>	Extracurricular Activities	Performance Index 1	Performance Index 2	Performance Index 3	Performance Index 4	0	~1200	~1100	~1150	~1050	1	~1100	~900	~950	~1000
Extracurricular Activities	Performance Index 1	Performance Index 2	Performance Index 3	Performance Index 4												
0	~1200	~1100	~1150	~1050												
1	~1100	~900	~950	~1000												
<p>Điểm trung bình của sinh viên ở từng nhóm có hoặc không tham gia hoạt động</p> <p>Đối với nhóm có tham gia hoạt động điểm trung bình sẽ cao hơn nhóm không tham gia do số lượng điểm mức 4 cao hơn mức 1 đã nêu ở thống kê trên. Nhìn chung, có hay không tham gia hoạt động có ảnh hưởng đến Perfomance Index nhưng không quá lớn, chỉ có ảnh hưởng đến số sinh viên ở mức 1 và 4</p>	 <table><caption>Data for Performance Index vs Extracurricular Activities</caption><thead><tr><th>Extracurricular Activities</th><th>Performance Index</th></tr></thead><tbody><tr><td>0.0</td><td>~54.7</td></tr><tr><td>1.0</td><td>~55.6</td></tr></tbody></table>	Extracurricular Activities	Performance Index	0.0	~54.7	1.0	~55.6									
Extracurricular Activities	Performance Index															
0.0	~54.7															
1.0	~55.6															

f. Phân tích đặc trưng Sleep Hours

<p>Thống kê số lượng sinh viên theo giờ ngủ</p> <p>Ta thấy được phân bố số lượng sinh viên của từng mức giờ ngủ khá đồng đều trong bộ dữ liệu, số lượng sinh viên có mức giờ ngủ 8h có chút nhỉnh hơn so với các mức còn lại.</p>	 <table><tr><th>Sleep Hours</th><th>count</th></tr><tr><td>4</td><td>1450</td></tr><tr><td>5</td><td>1450</td></tr><tr><td>6</td><td>1500</td></tr><tr><td>7</td><td>1500</td></tr><tr><td>8</td><td>1600</td></tr><tr><td>9</td><td>1450</td></tr></table>	Sleep Hours	count	4	1450	5	1450	6	1500	7	1500	8	1600	9	1450																					
Sleep Hours	count																																			
4	1450																																			
5	1450																																			
6	1500																																			
7	1500																																			
8	1600																																			
9	1450																																			
<p>Thống kê số lượng sinh viên theo giờ ngủ và thành tích ở các mức 1, 2, 3, 4</p> <p>Không có sự phân bố rõ rệt nào ở từng mức giờ ngủ, nhưng ta có thể thấy sự chênh lệch thành tích mức 1 và 4 ở mức giờ ngủ nhỏ nhất (4h) và mức giờ ngủ lớn nhất (9h). Tại mức 4h, số lượng sinh viên đạt điểm mức 1 rất cao so với mức 4 cũng như các mức khác. Tại mức 9h, số lượng mức 4 cao hơn các mức còn lại. Số giờ ngủ có sự ảnh hưởng đến Performance Index nhưng không quá rõ ràng.</p>	 <table><tr><th>Sleep Hours</th><th>1</th><th>2</th><th>3</th><th>4</th></tr><tr><td>4</td><td>400</td><td>350</td><td>350</td><td>320</td></tr><tr><td>5</td><td>380</td><td>350</td><td>380</td><td>350</td></tr><tr><td>6</td><td>400</td><td>380</td><td>350</td><td>380</td></tr><tr><td>7</td><td>380</td><td>350</td><td>380</td><td>380</td></tr><tr><td>8</td><td>420</td><td>420</td><td>420</td><td>420</td></tr><tr><td>9</td><td>350</td><td>380</td><td>350</td><td>400</td></tr></table>	Sleep Hours	1	2	3	4	4	400	350	350	320	5	380	350	380	350	6	400	380	350	380	7	380	350	380	380	8	420	420	420	420	9	350	380	350	400
Sleep Hours	1	2	3	4																																
4	400	350	350	320																																
5	380	350	380	350																																
6	400	380	350	380																																
7	380	350	380	380																																
8	420	420	420	420																																
9	350	380	350	400																																
<p>Điểm trung bình của sinh viên ở các mức giờ ngủ</p> <p>Tại mỗi mức giờ ngủ, ta tiến hành tính điểm trung bình của tất cả sinh viên có số giờ ngủ đó. Khi số giờ ngủ tăng thì nhìn chung điểm cũng tăng nhưng không ổn định, không phải tại mọi điểm có giờ ngủ cao hơn thì luôn có điểm cao hơn. Do đó, sự tác động của Sleep Hours đến Performance Index là có nhưng không ổn định và rõ ràng.</p>	 <table><tr><th>Sleep Hours</th><th>Performance Index</th></tr><tr><td>4</td><td>53.5</td></tr><tr><td>5</td><td>55.0</td></tr><tr><td>6</td><td>54.8</td></tr><tr><td>7</td><td>55.5</td></tr><tr><td>8</td><td>55.7</td></tr><tr><td>9</td><td>56.5</td></tr></table>	Sleep Hours	Performance Index	4	53.5	5	55.0	6	54.8	7	55.5	8	55.7	9	56.5																					
Sleep Hours	Performance Index																																			
4	53.5																																			
5	55.0																																			
6	54.8																																			
7	55.5																																			
8	55.7																																			
9	56.5																																			

g. Phân tích đặc trưng Sample Question Papers Practiced



5. Xây dựng mô hình dự đoán chỉ số thành tích sử dụng toàn bộ 5 đặc trưng

Để tiện cho việc viết phương trình ta có các ký hiệu sau

- HS: Hours Studied
- PS: Previous Scores
- EA: Extracurricular Activities
- SH: Sleep Hours
- SQ: Sample Question Papers Practiced

Phương trình hồi quy tuyến tính khi sử dụng 5 đặc trưng có dạng:

$$PI = w_0 + w_1HS + w_2PS + w_3EA + w_4SH + w_5SQ$$

Các bước thực hiện:

- Từ tập dữ liệu train, ta tiến hành tạo ma trận đầu vào X_{m_train} với cột đầu tiên có tất cả giá trị là 1 và 5 cột tiếp theo là giá trị của 5 đặc trưng, y_{m_train} với 1 cột là giá trị PI (Performance Index)
- Sử dụng lớp `OLSLinearRegression` để tính hệ số hồi quy của model và làm tròn 3 chữ số thập phân bằng hàm `numpy.round()`
- Từ tập dữ liệu test, ta tiến hành tạo ma trận đầu vào X_{m_test} với cột đầu tiên có tất cả giá trị là 1 và 5 cột tiếp theo là giá trị của 5 đặc trưng, y_{m_test} với 1 cột là giá trị PI (Performance Index)
- Gọi hàm `predict()` của model vừa tạo trên để dự đoán giá trị đầu ra dữ liệu y_hat
- Sử dụng hàm `mae()` với đối số là y_{m_test} và y_hat để tính sai số tuyệt đối trung bình

Kết quả:

- Công thức hồi quy:

$$PI = -33.969 + 2.852HS + 1.018PS + 0.604EA + 0.474SH + 0.192SQ$$

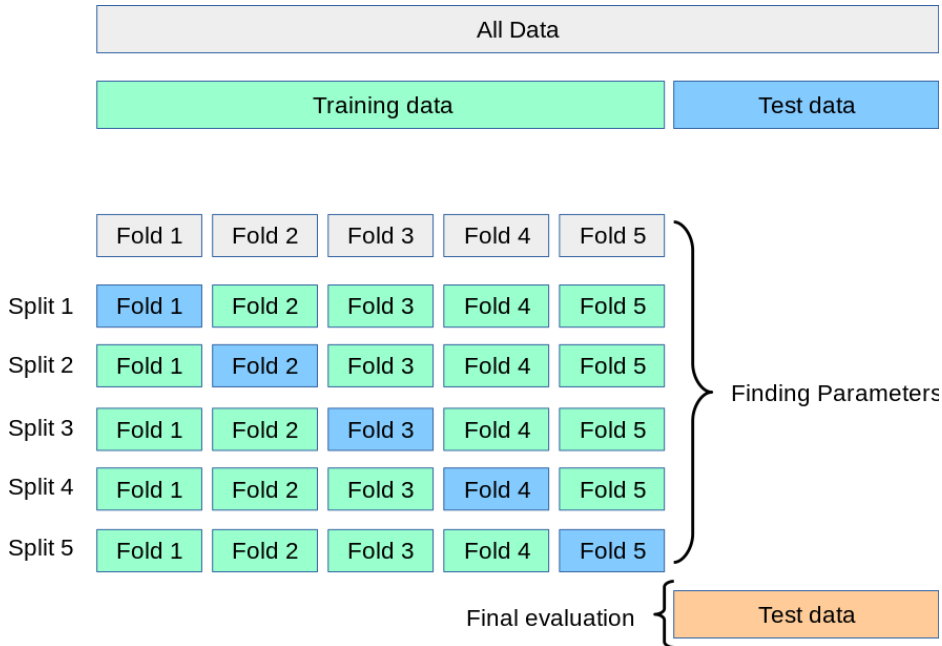
- MAE trên tập test:

$$MAE = 1.5956$$

- Nhận xét: Kết quả hợp lý với những phân tích của phần phân tích dữ liệu, cả 5 đặc trưng đều có những ảnh hưởng (nhiều hoặc ít) đến Performance Index, do đó model sử dụng cả 5 đặc trưng cho kết quả khá tốt ở tập test.

6. Xây dựng mô hình sử dụng 1 đặc trưng, tìm mô hình cho kết quả tốt nhất

K-fold validation [4]: phương pháp đánh giá độ chính xác của model trên một tập dữ liệu. Phần dữ liệu D_{train} sẽ được chia thành k phần, ta chọn 1 phần dữ liệu làm D_{val} và $k-1$ phần dữ liệu còn lại làm D'_{train} . Các phần dữ liệu sẽ thay nhau đảm nhận vị trí D_{val} cho đến khi tất cả chúng đều đã làm D_{val} . Ta tiến hành train trên tập D'_{train} và test trên tập D_{val} . Ở mỗi lần test, ta có được MAE tương ứng và kết quả của thuật toán sẽ là trung bình MAE.



Phương trình hồi quy tuyến tính khi sử dụng một đặc trưng duy nhất:

$$PI = w_0 + w_1 feature$$

Các bước thực hiện:

- Tiến hành xáo trộn các dòng của tập dữ liệu ban đầu thành tập D
- Lập k lần để tiến hành xác định k tập D_{val} và D_{train} .
- Ở mỗi lần lập, sau khi có tập D_{val} và D_{train} ta tiến hành lập 5 lần để train cho cả 5 đặc trưng và tính mae (tức ta có 2 vòng lặp: lặp k lần và lặp 5 lần lồng trong mỗi lần lặp k)
- Ở mỗi lần lập để train cho một đặc trưng, Xm_{train} , Xm_{val} là ma trận đầu vào của lớp `OLSLinearRegression` với cột đầu tiên có giá trị toàn 1, cột thứ 2 là giá trị của đặc trưng ở D_{train} , D_{val} .
- Gọi hàm `fit()` của `OLSLinearRegression` để tính giá trị hệ số hồi quy. Gọi `predict()` để dự đoán giá trị y_{hat} và `mae()` để tính sai số.

- Khi chạy xong vòng lặp, ta có ma trận maes với 5 dòng (5 đặc trưng) và k cột (k lần train và test cho ra k giá trị mae). Gọi hàm `numpy.average()` để tính mae trung bình của từng đặc trưng.

Kết quả:

STT	Mô hình với 1 đặc trưng	MAE
1	Hours Studied	15.448
2	Previous Scores	6.618
3	Extracurricular Activities	16.193
4	Sleep Hours	16.186
5	Sample Question Papers Practiced	16.184

Nhận xét: Từ bảng kết quả trên, ta thấy Previous Scores là đặc trưng cho kết quả tốt nhất. Kết quả trên là hợp lý vì từ biểu đồ tương quan cho đến các biểu đồ thống kê đều cho thấy PS là đặc trưng quan trọng ảnh hưởng ổn định đến Performance Index, các đặc trưng còn lại vẫn có ảnh hưởng nhưng chưa ổn định nên sai số lớn hơn.

Vậy phương trình hồi quy tuyến tính sử dụng đặc trưng Previous Scores: $PI = w_0 + w_1PS$

Các bước thực hiện:

- Từ tập dữ liệu train, ta tiến hành tạo ma trận đầu vào `Xm_train_2b` với cột đầu tiên có tất cả giá trị là 1 và cột tiếp theo là giá trị của Previous Scores, `ym_train_2b` với 1 cột là giá trị PI (Performance Index)
- Sử dụng lớp `OLSLinearRegression` để tính hệ số hồi quy của model và làm tròn 3 chữ số thập phân bằng hàm `numpy.round()`
- Từ tập dữ liệu test, ta tiến hành tạo ma trận đầu vào `Xm_test_2b` với cột đầu tiên có tất cả giá trị là 1 và cột tiếp theo là giá trị của Previous Scores, `ym_test_2b` với 1 cột là giá trị PI (Performance Index)
- Gọi hàm `predict()` của model vừa tạo trên để dự đoán giá trị đầu ra dữ liệu `y_hat_2b`
- Sử dụng hàm `mae()` với đối số là `ym_test_2b` và `y_hat_2b` để tính sai số tuyệt đối trung bình

Kết quả:

- Công thức hồi quy:

$$PI = -14.989 + 1.011PS$$

- MAE trên tập test:

$$MAE = 6.5443$$

- Nhận xét: Kết quả MAE khi đo trên tập test gần bằng khi đo trên tập D_{val} của k-fold cross validation. Khi sử dụng duy nhất một đặc trưng, kết quả MAE không quá tốt do các đặc trưng các cũng có tương quan đến Performance Index, việc bỏ qua các đặc trưng khác sẽ cho kết quả sai số lớn.

7. Xây dựng/thiết kế mô hình, tìm mô hình cho kết quả tốt nhất

Các bước thực hiện:

- Tập D với bộ dữ liệu được xáo trộn ở câu 2b sẽ được dùng lại ở câu 2c
- Ta tiến hành train cho từng model, không train cho tất cả cùng lúc như câu 2b
- Mỗi lần chạy k-fold validation, thuật toán sẽ được lặp k lần
- Tại mỗi lần lặp xác định D_{val} và D_{train}
- Từ đó có thể xác định Xm_{val} , Xm_{train} tùy vào mô hình
- Kết quả là trung bình MAE
- Sau khi train tất cả các model, chọn ra model có MAE trung bình thấp nhất và train với tập train lại rồi test với tập test

a. Mô hình 1

Từ biểu đồ tương quan, ta có thể thấy Hours Studied và Previous Scores là 2 đặc trưng có chỉ số tương quan cao nhất và thông qua phân tích biểu đồ cũng cho thấy 2 đặc trưng có ảnh hưởng rõ rệt đối với Performance Index, do đó ta chọn model sử dụng 2 đặc trưng HS và PS

$$PI = w_0 + w_1HS + w_2PS$$

Kết quả

- MAE trung bình: 1.816
- Nhận xét: Kết quả khá tốt khi chỉ sử dụng 2 đặc trưng có chỉ số tương quan lớn nhất. Nhưng các đặc trưng khác vẫn có tương quan với Performance Index, ta xem xét vài mô hình sử dụng cả 5 đặc trưng với sự chuẩn hóa khác nhau.

b. Mô hình 2

Tuy các đặc trưng còn lại có chỉ số tương quan thấp nhưng chúng cũng có sự tác động ảnh hưởng đến Performance Index thông qua phân tích trên biểu đồ, do đó có sự cân nhắc khi sử dụng cả 5 đặc trưng. Qua quan sát biểu đồ, thấy được đặc điểm của hai đặc trưng SH và SQ khá giống nhau, khi chúng chỉ có sự ảnh hưởng rõ ràng đến Performance Index ở các mức giá trị cao nhất và thấp nhất. Do đó, ta tạo ra đặc trưng mới là tổng của SH và SQ

$$PI = w_0 + w_1HS + w_2PS + w_3EA + w_4(SH + SQ)$$

Kết quả:

- MAE trung bình: 1.653
- Nhận xét: Do SH và SQ có tính chất tương đồng, sử dụng đặc trưng mới là tổng của chúng cùng 3 đặc trưng còn lại cho kết quả tốt hơn khi chỉ sử dụng 2 đặc trưng HS và PS

c. Mô hình 3

Như đã nêu ở trên, sự tác động của SH và SQ lên Performance Index là không ổn định. Điều đó thể hiện ở biểu đồ điểm trung bình, khi giá trị của SH và SQ tăng thì chưa chắc rằng Performance Index cũng sẽ tăng. Sự phân bố số lượng điểm mức 1 và 4 chỉ rõ rệt ở các giá trị min và max của SH và SQ. Ta nghĩ đến phương pháp bình phương giá trị của SH và SQ lên nhằm tăng sự chênh lệch giữa các giá trị. Khi đó, giá trị càng gần min sẽ càng nhỏ và giá trị càng gần max sẽ càng lớn. Càng gần min thì sẽ có xu hướng thành tích thấp hơn, gần max thì có xu hướng thành tích cao hơn.

$$PI = w_0 + w_1HS + w_2PS + w_3EA + w_4SH^2 + w_5SQ^2$$

Kết quả:

- MAE trung bình: 1.625
- Nhận xét: Kết quả nhỉnh hơn với mô hình 2 khi tăng sự chênh lệch giữa các giá trị SH và SQ.

d. Huấn luyện my_best_model

STT	Mô hình	MAE
1	Sử dụng 2 đặc trưng (HS, PS)	1.816
2	Sử dụng đặc trưng HS, PS, EA và một đặc trưng mới là tổng của SH và SQ	1.653
3	Sử dụng đặc trưng HS, PS, EA và bình phương SH và SQ	1.625

Mô hình 3 cho kết quả MAE bé nhất, do đó chọn mô hình 3 để huấn luyện trên tập train

Kết quả:

- Mô hình hồi quy tuyến tính:

$$PI = -32.222 + 2.853HS + 1.018PS + 0.609EA + 0.036SH^2 + 0.02SQ^2$$

- MAE trên tập test:

$$MAE = 1.6$$

- Nhận xét: Kết quả trên tập test cho ra gần bằng kết quả khi sử dụng k-fold cross validation cho thấy sự ổn định của mô hình này.

III. Tài liệu tham khảo

- [1] pandas, "pandas - Python Data Analysis Library," [Online]. Available: <https://pandas.pydata.org/docs/reference/>.
- [2] matplotlib, "matplotlib.pyplot," [Online]. Available: https://matplotlib.org/3.5.3/api/_as_gen/matplotlib.pyplot.html.
- [3] thebmj, "Correlation and regression," [Online]. Available: <https://www.bmj.com/about-bmj/resources-readers/publications/statistics-square-one/11-correlation-and-regression>.
- [4] geeksforgeeks, "Cross Validation in Machine Learning," [Online]. Available: <https://www.geeksforgeeks.org/cross-validation-machine-learning/>.